Semi-Automated Analysis of Electronic Commerce Application Specifications

Alan Abrahams and Naila Mimouni
University of Cambridge Computer Laboratory
{asa28, nkm23}@cl.cam.ac.uk

Abstract

This paper examines the issue of automating business analyses. It aims at providing a direct mapping from business process specifications to software implementations. This involves the analysis, modelling and interrogation of the business occurrences and contractual provisions that drive workflow applications. Although crucial, semi-automated analysis and executable specification of electronic commerce application user requirements has been somewhat neglected. In order to address this matter, we have developed a novel occurrence-based development approach and execution infrastructure to create executable and queryable specifications for electronic commerce applications. In this paper, we outline a set of guidelines that can be used to expose occurrences from appearances of certain words and word forms in English-language specifications. The analysis output can be used for software implementation. While this does not provide the final solution, it represents a significant step towards the ultimate goal of automating contract analyses.

1 Introduction

To provide a more direct mapping of business process specifications to software implementations, the next generation of enterprise workflow systems must move away from a procedural execution style and towards an *event-driven model* that monitors and controls the business process in accordance with a periodically changing set of stored business contracts, intra-organizational policies, and legislative requirements.

A crucial, and hitherto neglected, aspect of electronic commerce application development techniques and tools is the analysis, modelling, storage, and interrogation of the business occurrences and contractual provisions that drive workflow applications. Extant systems for event monitoring, business rules, policy-based management, contracting, and workflow execution do not directly represent, store, enact and enforce the subtle and often conflicting contractual and regulatory provisions contained in business requirements specifications. Explicit treatments of fundamental legal conceptions such as obligations, permissions, and powers are absent from conventional software. Importantly, current approaches lack guidelines that allow analysts to transform English-language specifications of contracts, policies, laws, and regulations into a structured form suitable for direct input into an implementation environment. This makes seamless passage through the system development life cycle a hard problem.

The transition from specification to implementation cannot be completely systematized; the natural language processing and artificial intelligence problem is, at present, very difficult. Problems with the processing of software requirements specifications in unrestricted natural language include failure of the lexicon, assignment of multiple parses, or failure of semantics [OM96]. We do not therefore propose to address natural language processing issues. Rather, we seek to pay some attention to how contract structure and workflow occurrences are expressed in natural language, with the goal of providing helpful insights to aid in capturing the essentials of a business specification. We hope to illustrate that the transition from analysis to implementation in an occurrence-based development style may be guided by the application of an explicit set of rules, and there is consequently less reliance on the uncodified experience of an expert. The use of written rules makes the process of translation from English-language specification to implementation more systematized, disciplined, and repeatable. Accordingly, we describe a set of guidelines that can be used by an analyst to undertake a formal analysis of business contracts and user requirements specifications. These rules may be used to expose occurrences from appearances of certain words and word forms in English-language specifications. The guidelines are labelled and applied to worked examples from an application scenario that we have developed. The envisaged output of applying these rules is not procedural code, but rather contract structure that is actionable and can be monitored.

In Section 2, we review some related work. Section 3 describes the application scenario that we use to illustrate our guidelines. In Section 4, we propose various techniques that can be employed to expose occurrences in business process application specifications. We look, in turn, at domain-specific occurrences (§4.1), and occurrences of selection (§4.2), quantification (§4.3), sorting and comparison (§4.4), prescription (§4.5), and description (§4.6).

2 Related Work

As previously mentioned, the current state-of-the-art in natural language understanding technology does not allow for parsing and complete automation of business laws and requirements. Only a few researchers have made a significant attempt at the problem.

Previous work in the area of natural language requirements analysis includes the KISS approach and associated Grammalizer tool [HvdVH97], which help analysts to derive a conceptual model from a textual domain description.

Lloyd [Llo2000, BLM2001] has built a prototype that aims at translating a controlled English into machine-interpretable access control expressions. The system includes an interactive component allowing the user to correct their

policy statements. However, it deals only with a limited class of policy statements and does not resolve conflicts. Michael, Ong, and Rowe [MOR2001] emphasize that being able to quickly translate a natural language specification of security policy into a formal logic would be useful as policy changes frequently, policy bases can be large, and the relationships between policies may be complex. Ideally, they argue, policy should be stored in a computational form in a centralized, searchable, and updateable repository. The authors describe their natural-language-input-processing tool: this is merely a part-of-speech tagger which maps natural language sentences expressing security policies to an object-oriented schema and allows queries against the schema. No mechanised interpretation and enforcement of the output is described.

In this paper, we provide a set of guidance rules, which can be employed by an analyst to expose salient occurrences in English user requirements documents, such as business contracts, policies, and legislation. Our previous investigations into event semantics led to the definition of a database schema for the representation of these variable-attribute occurrences [AB2001b], paving the way to interrogation and execution of stored e-commerce application specifications. Our prototype software, EDEE, provides a platform-independent active wrapper [AB2001c], which allows us to record, reason about, and enact contractual provisions [AB2001d, AB2002a]. In earlier work, we demonstrated a novel query storage and coverage determination mechanism, which allows contract performance monitoring and facilitates dynamic consistency checking of contracts against policies [AEB2002a, AEB2002b]. EDEE's coverage determination component has been supplied to researchers at the University of Aachen, where it is being used for the monitoring of contracts in a business-to-business electronic marketplace [Sta2002]. In related papers, we have also proposed a new model of the life and times of identified and situated norm instances [AB2002b, AK2002]. The model is used in our contract-driven and legislation-aware workflow automation approach [Abr2002], to support conflict resolution [AEB2002c].

3 Application Scenario

We now introduce an application scenario to clarify the problems we address and illustrate the plausibility of our implementation.

SkyHi Builders is a construction company. Steelmans Warehouse is a supplier of high-grade steel. SkyHi, having recently won a tender to build a new office block, enters into a contract with Steelmans. An excerpt appears as follows:

Contract between SkyHi and Steelmans entered into on 1st August 2001

…

"steel" shall mean low-carbon steel of the type Fe360 (Euro-Norm 10025) in sheets with dimensions 1600 x 400 x 5.0 mm, with thickness tolerance ± 0.040 mm on a single sheet.	**Clause D.1**
SkyHi must pay Steelmans $25,000 before 1st September 2001.	**Clause C.1**
Steelmans must deliver 10 tons of steel before 1st October 2001.	**Clause C.2**
SkyHi has the right to return the steel within 30 days.	**Clause C.3**
In the event of a return in terms of Clause C.3 above, Steelmans shall refund SkyHi the amount paid.	**Clause C.4**

…

In addition, SkyHi has the following internal organizational policies:

SkyHi Risk Management Procedures

…

Clerks may not buy steel.	**Clause P.1**
Employees older than 25 may buy steel.	**Clause P.2**
Payments of more than $10,000 to suppliers are prohibited.	**Clause P.3**

…

And SkyHi finds itself subject to the following provisions of legislation:

Commercial Trade Act

…

An obligation is fulfilled when all obliged occurrences have happened.	**Clause L.1**
An obligation is violated if it is after the deadline and some obliged occurrences have not happened.	**Clause L.2**
Following successful instigation of the prescribed procedure for claiming compensation, damages for violation of an obligation must be paid, by the liable party, to the party entitled to compensation.	**Clause L.3**

…

SkyHi wishes to store the provisions of their contracts and internal business policies, and the legal regulations to which SkyHi is subject, in a database, so that the provisions can be used to guide the behaviour of their computer and human-activity systems. Scripting the system with procedural code is not an option: the sequence of SkyHi's business processes is not static and they do not wish to employ a programmer to sift through and change procedural code to reflect the frequent

alterations in contracts, policies, and regulations. SkyHi would like the human and software components in their system to consult the database in order to determine what to do next in the light of a dynamically changing set of provisions.

4 Occurrence-based Analysis Guidelines

This section details a set of guidelines that can be used by an analyst to undertake a formal analysis of business contracts and user requirements specifications. It illustrates basic rules that may be used to expose occurrences from appearances of certain words and word forms in English-language specifications. The input to our process is a set of textual business contracts, policies, and regulations, provided by management, including user requirements documents provided by a business analyst (the application scenario in Section 3 is an example of this). These define what the various role-players in the system *can* and *must* do under various circumstances, as well as what the computerized system itself *can* and *must* do under various circumstances. The envisaged output of applying these rules is not procedural code, but rather contract structure that is actionable and can be monitored. The analysis output can then be used for direct software implementation.

We treat an **occurrence** as being an instance of a specific relationship type or association type that exists between entities, at a moment in time or over an interval in time. An occurrence may be an event, a state, or a process. For instance, we would treat *buying*, *owning*, *approving*, *being-obliged*, and *being-prohibited* as occurrence types. An occurrence type (such as *buying*) may have multiple occurrence instances. Each occurrence instance has role-players acting in a role in the occurrence: an occurrence of *buying* typically has at least participants in the roles *buyer*, *seller*, *sold item*, and *purchase price*[1].

In each of the sections that follow, we propose various techniques that can be employed to expose occurrences in business process application specifications.

4.1 Domain-Specific Occurrences

Identification of domain-specific occurrences from an English-language specification may proceed through a search for explicit verbs, deverbative nouns and roles indicating underlying occurrences. Explicit verbs indicate occurrences of events or states and can be detected through a number of means including indicative suffixes, or consultation of a lexicon.

Indicative suffixes such as –ing, –s, and –ed on words often point to the existence of occurrences whose type is the gerund[2] form of the word. Occurrences are also often indicated by non-modal auxiliaries: 'is', 'was', 'being', 'been', 'are', 'were', 'will', 'have', 'has'. **Guideline 1**

Examples:
By Guideline 1, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Occurrences of Type
Owning		Owning
Owns		Owning
Owned		Owning
Is/Has overdrawn		Being overdrawn / overdrawing

Counterexamples: The ending –s can indicate the plural form of a noun. e.g. 'dogs'. **Exceptions**

Deverbative nouns are noun forms of verbs and can reveal underlying occurrences.

Indicative suffixes such as as -ion, -ment, -ent, -ure, -ance, -ence, -ancy, -ency, -ing, -al, -y, or –age in a deverbative noun often point to the existence of occurrences whose type is the gerund form of the deverbative noun. **Guideline 2**

Examples:
By Guideline 2, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Occurrences of Type
Registration		Registering
Acceptance		Accepting
Examples from Application Scenario:		
Tolerance	(Clause D.1)	Tolerating
Payment	(Clause P.3)	Paying
Violation	(Clause L.3)	Violating

Counterexamples: Some words ending in one of the suffixes listed above do not point to an occurrence. e.g. 'distance'. **Exceptions**

[1] Inanimate 'participants', such as *purchase price*, might be more easily thought of as attribute values for attributes of the occurrence.

[2] A canonical form ending in –ing.

Parsons [Par90], in his theory of event and state semantics, advises that some suffixes may be indicative of instances of states.

Suffixes such as such as –ness, -ship, –hood, and –ly may indicate underlying occurrences of states or events.	**Guideline 3**

Examples:
By Guideline 3, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Occurrences of Type
Illness		Being ill (state)
Allegedly		Alleging (event)

Examples from Application Scenario:

Thickness	(Clause D.1)	Being thick

More specifically, 'being thick', indicates an occurrence of measuring where dimension_measured is 'thickness' (see §4.3)

Counterexamples: 'harness'.	**Exceptions**

There are also a number of ways in which occurrences of possessing are indicated in English.

The ending –s' or –'s, the preposition 'of', and the possessive pronouns 'his', 'her', 'their', 'our', and 'its' may indicate occurrences of possessing, having, or owning.	**Guideline 4**

Word forms denoting the names of roles held by participants in an occurrence often indicate underlying domain-specific occurrences.

Indicative suffixes such as –er, –or, –ar, –ee, –ant, –ent, –ed, -d, –en, or –yst on an English word typically denote role names, and often point to the existence of occurrences whose type is the gerund form of the role name.	**Guideline 5**

Examples:
By Guideline 5, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Occurrences of Type
Employer		Employing
Director		Directing
Applicant		Applying
Resident		Residing
Analyst		Analysing

Examples from Application Scenario:

Paid	(Clause C.4)	Paying
Employee	(Clause P.2)	Employing
Supplier	(Clause P.3)	Supplying
Entitled[3]	(Clause L.3)	Being entitled

Counterexamples: 'coffee', 'detergent', …	**Exceptions**

Exposing underlying occurrences from role names is an important step in requirements elicitation, as it allows the analyst to identify and record information about these associations. For example, the identification of an occurrence of *employing* implies recording the holder of the employee role (which is explicit in the specification in our application scenario), the holder of the employer role, and the start- and end- dates of employment (which are implicit in the specification). Also, responsibilities and privileges are typically associated with each identified role. During requirements elicitation, an analyst can uncover the norms associated with each role by using templates such as '[role-name] must ...', '[role-name] must not ...', and '[role-name] can ...'. For instance '[applicants] can register for the conference by completing the registration form before the deadline'.

4.2 Selection Occurrences (Queries)

Modifiers in English are used to select entities based on qualification (matching or conformance with recorded criteria).

Modifiers or qualifiers are often indicated by (explicit or implicit) 'that' or 'which' and imply the existence of a query covering a set of individuals, and the use of that query for selecting, at a certain time, a particular set or sets.	**Guideline 6**

[3] More specifically, 'being entitled' or 'having a right', implies an occurrence of someone else *being obliged*.

Examples from Application Scenario:
By Guideline 6, (Provenance)

(Evidence) Appearance of Word	… indicates …	(Consequence) Resulting query
Employees (that are) older than 25 (Clause P.2)		`[participants in role [=employee] in [occurrences of [employing]]] intersection [participants in role [=aged] in occurrences of [being_aged] where [participants in role [=age] are [>25]]`
Payments (that are) more than $10,000 (Clause P.3)		`occurrences of [paying] where [>10,000] is [=paid_amount]`

Various English words or symbols may indicate specific set-operators. **Guideline 7**

Examples:
By Guideline 7, (Provenance)

(Evidence) Appearance of Word / Symbol	… indicates …	(Consequence) Set Operator
'and', 'also', 'with', 'that', 'which', and adjectives or adjectival clauses e.g. 'low-carbon [adj.] steel'; 'steel that is low in carbon'; 'steel with low-carbon content')		intersection e.g. steel ∩ low-carbon things
'and', 'or', comma (','), semi-colon (';'), bullet (list). e.g. customers and employees		union e.g. customers ∪ employees
'but', 'not', 'except', 'excluding', 'apart from', 'besides', 'without', 'with the exception of', 'save', 'however', 'although' e.g. 'customers but not gold customers'		difference e.g. customers – gold customers

Words used for discourse deixis[4] – such as 'above', 'below', 'earlier', 'later', 'this', 'here', 'there', 'previous', 'following', 'next', and cross-references to documents or chapter and section headings – may indicate the existence of a query that selects labelled utterances or provisions. Typically, provisions are selected so that they may be voided during conflict resolution [AEB2002c], or in order to choose which clauses specify all-things-considered obligations [Abr2002]. **Guideline 8**

Examples:
By Guideline 8, (Provenance)

(Evidence) Appearance of Word	… indicates …	(Consequence) Query
'above'		a query that selects all clauses that appear above the current clause in the current document

4.3 Quantification Occurrences

Counting, usually by storing and executing `count` queries, is the simplest form of quantification occurrence. Other forms of quantification include measuring by observing or computing.

The English cardinals ('one', 'two', etc.) and the quantifiers 'a' / 'one', 'none' / 'no' / 'not' (and negation affixes such as un–, il–, non–, im–, in–, –less, –free), 'some', 'few', 'multiple', 'many', 'most', 'each' / 'all' / 'every', 'only', 'low', and 'high', may imply counting or measuring, and can also indicate implicit prohibitions (§4.5) or powers (§4.6). With vague quantifiers, a specific convention is typically applied (e.g. 'few, according to clause *x*' is '<3'; 'low, according to clause *y*' is '≤ 10'). The exact convention used should be made explicit to avoid fuzziness in the contract. **Guideline 9**

[4] 'Discourse deixis' is a term used in linguistics for expressions that point to other utterances in a verbal or textual discourse.

Examples:
By Guideline 9, (Provenance)

(Evidence) Appearance of Word	… indicates …	(Consequence) Occurrences of Type
No		counting, with count = 0
All		counting (where two counts are equal)
three management signatures are needed		counting, of occurrences of signing by managers
only managers possess company credit cards		'zero non-managers possess company credit cards' or "`count (occurrences of possessing with credit cards in role [=possessed] and (universe minus managers) in role [=possesser])`; if the result of the counting exceeds zero, the counted items violate policy"
every manager possesses a company credit card		'zero managers not possess company credit cards' or "`count (managers that do not participate in occurrences of possessing with credit cards as the possessed)`; if the result of the counting exceeds zero, the counted items violate policy"

Examples from Application Scenario:

all	(Clause L.1)	counting, where `count of actual occurrences = count of obliged occurrences`
some	(Clause L.2)	counting, where `count of actual occurrences < count of obliged occurrences`
low carbon	(Clause D.1)	measuring, of carbon content, and comparing to threshold

'Adverbs of frequency', which stand in the place of the usual quantifiers, may show that occurrences are being quantified over. **Guideline 10**

For instance:

Regular Quantifier	… becomes …	Frequency Adverb (Quantifier over Occurrences)
No		never / at no time / not once
All		always
some		sometimes / occasionally
few		rarely / seldom / almost never / hardly ever
many		often / regularly
most		usually / normally / almost always
one		once
two		twice

Examples:
By Guideline 10, (Provenance)

(Evidence) Appearance of Word	… indicates …	(Consequence) Occurrences of Type
Steelmans never delivered on time		counting, with (count of on-time occurrences of delivering by Steelmans) = 0

The appearance of a numeral, or a unit of measure, in a specification typically denotes that some form of counting or measuring has occurred or must occur. **Guideline 11**

Examples from Application Scenario:
By Guideline 11, (Provenance)

(Evidence) Appearance of Word	… indicates …	(Consequence) Occurrences of Type
1600 x 400 x 5.0 mm (Clause D.1)		measuring: at least three measuring occurrences, where steel is item_measured and length, width, and thickness are the dimension_measured

Occurrences of measuring should take a quantity (number), a unit of measure (e.g. 3 metres, 5 managers), an item measured, and a dimension (e.g. height, width). For instance, an occurrence of measuring the thickness of a delivered sheet of steel would be denoted thus:

measuring1 (occurrence instance)

(role)	(participant)
item_measured:	sheet4
quantity_measured:	5.02
unit_of_measure:	mm
dimension_measured:	thickness
measurer:	Bob (Quality Controller 3)

Presuming the sheet also measured 1600mm in length, and 400mm in width, and had a computed carbon content of less than 0.1% (the maximum prescribed by Euro-Norm 10025 for low-carbon steel) this sheet would – at least temporarily – be within the quality specifications of the clause, and would therefore count as being 'steel', in terms of Clause D.1.

4.4 Sorting and Comparison Occurrences

Comparitives, superlatives, ordinals and some function words may imply occurrences of sorting (i.e. ordering or sequencing) or alternatively merely occurrences of comparing. Every occurrence of sorting includes occurrences of comparing using some comparator function.

Guideline 12

The existence of implicit occurrences of *sorting* and *comparison* may be indicated variously by:

- ordinals like 'first', 'second', 'third', …, 'last'
- suffixes such as –er, –st, –nd, –rd, and –th (e.g. 'richer', 'wealthiest')
- prefixes like pre–, post–, and suc–
- lexical items such as 'than', 'too', 'exceeds', 'excess', 'enough', 'before', 'after', 'prior', 'different', 'same', 'more', 'most', 'less', 'least', 'worse', and 'worst'.

Examples:

By Guideline 12, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Occurrences of Type
The lowest cost		sorting by cost
before 1st October		comparing temporal order
too long / long enough		comparing to a deadline or threshold

Examples from Application Scenario:

more than $10,000 (Clause P.3)	comparing to threshold measured in units of dollars
older than 25 (Clause P.2)	comparising to an age threshold measured in years

Ranges involve the use of quantities that act as upper and/or lower limits (minima or maxima).

Guideline 13

Ranges are often signalled by means and tolerance levels (e.g. 12pm ± 30 minutes), or by the prepositions and words: 'between', 'in', 'within', 'above', 'below', 'from… to…', 'on', 'during', 'more than', 'less than', 'maximum', 'minimum', 'limit', or 'bounds').

Examples from Application Scenario:

± 0.040 mm (Clause D.1)	comparing to upper and lower bounds

Sorting may be accomplished by explicitly tagging items with their predecessor and successor, the comparator function used and the time of comparison. Occurrences of *preceding*, in the form 'x *preceding* y in comparison order z, at time t', can be stored to capture the sequence.

The nature of the comparator function (i.e. comparison *metric* used) is important, and often needs to be made explicit. For instance, in 'the wealthiest individual' there is clearly some sorting of individuals, but the comparison function used may take into account the cash reserves of the individual, their 'paper wealth' in terms of shares, property valuations, or other criteria. It would not be contradictory for an individual to rate first in a comparison of cash wealth, but to rate lower down in the scale for a comparison of paper wealth done at the same time.

4.5 Normative (Prescriptive) Occurrences

Prescriptive policies define who can or must (or cannot or must not) do what, to what, and when; that is, they prescribe the behaviour of role-players in the system. Obligations (occurrences of *being-obliged*) associated with a named role are the responsibilities of the role – e.g. what the user *must* do. Authorisations (permissions and prohibitions) associated with a role are the privileges of the role – e.g. what the user *can* do.

Guideline 14

Appearances of the modal auxiliaries 'can', 'may', 'shall', 'must', 'has/have to', 'need to', 'should', 'could', 'would', 'ought', and 'will', 'has/have the right/authority to', 'is/are entitled to', or negations of these ('cannot', 'may not', etc.) indicate occurrences of *permitting*, *prohibiting*, or *being-obliged*. Likewise for the suffixes -able and -ible. These indicators may also, or alternatively, indicate the existence of a function [AB2002a, AEB2002b, Abr2002], which encodes a *power or liability*, or *disability or immunity*.

Examples:
By Guideline 14, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	*(Prima facie)* Occurrences of Type
Acceptable		*permitting* to accept / *power* to bring about an occurrence of *accepting*
Examples from Application Scenario:		
SkyHi must pay Steelmans (Clause C.1)		*being-obliged* to pay
Clerks may not buy steel (Clause P.1)		*prohibiting* buying *legal disability* of clerks to buy
SkyHi has the right to return the steel within 30 days (similarly "The steel is returnable within 30 days") (Clause C.3)		*power* to bring about an occurrence of *returning* in terms of that clause
"steel" shall mean … (Clause D.1)		a *power* to bring about an occurrence of *being-steel* for any item that meets the criteria
party entitled to compensation (Clause C.3)		another party *being-obliged* to do *compensating*

Often, 'must', 'shall', 'have to', 'will', or 'ought' indicate the existence of an obligation: 'SkyHi must (similarly shall/have-to/will/ought-to) pay Steelmans $25,000' may indicate that 'SkyHi *is obliged* to pay Steelmans $25,000'. Similarly, 'can' or 'may' regularly denote permission: 'SkyHi can (may) distribute steel in the East Anglia region' might have the intended reading 'SkyHi *is permitted* to distributed steel in the East Anglia region'. Finally, 'must not' could be read as a prohibition, as in 'Steelmans must not supply to other distributors' which might be read 'Steelmans *is prohibited* from supplying to other distributors'. Caution must, however, be exercised. Regarding every appearance of the word 'must' or 'shall' as implying an obligation is naïve. Consider that 'Managers must sign purchase orders' may be intended to mean 'anyone other than a manager is prohibited from signing purchase orders', and does not necessarily imply that managers are obliged to sign purchase orders.

It is also important to realize that in some contexts modal auxiliaries may not refer to legal concepts such as obligation at all. Instead (or in addition), they might indicate occurrences of *predicting*, *expecting*, or *intending* ('It will be delivered tomorrow'); *requesting* or *suggesting* ('Can you deliver ten tons of steel tomorrow?'); *offering*, *inviting to treat*[5], *volunteering*, or *accepting* ('You can store it in our warehouse'); or *being physically able* ('I can pay you tomorrow'). The distinction between permission and practical ability has been pointed out by Jones and Sergot [JS96]. The difference between predicting and promising has been explored in the literature on speech acts [Aus76, Sea69] and agent communication languages (ACLs); for instance, in the work on a Formal Language for Business Communication (FLBC) [KM97]. FLBC extends the simple sentences treated by other authors on event-semantics [Dav80, Par90], with sentences containing embedded propositional content. Such sentences include '*promising* that …', '*requesting* that …', and other examples prevalent in business communications. In each case, propositional content (e.g. 'I pay') is bracketed or 'quoted' within a sentence that indicates attitude[6] (e.g. 'I promise that [I pay]'). FLBC's contribution is in formalizing the suggestions of speech act theorists [Aus76, Sea69], and in applying these ideas to deontic reasoning and business messaging.

4.6 Conventional (Descriptive) Occurrences

Contracts and specifications commonly encode legal powers to bring about certain occurrences. Provisions use particular names or classifications for items fitting certain criteria. These are conditions that must be met in order for a naming to be legal in terms of some set of norms.

As is evident from Clause D.1, defining 'steel', in our application scenario, a particular word may cover different items according to different clauses. Consider that an item given the identifier `17986` may colloquially be called 'steel' because it is silver and shiny, but may not be 'steel, according to Clause D.1' because it either does not conform to the requirements of Euro-Norm 10025, or it does not meet the specifications constraining its allowable dimensions in terms of the clause. Steelmans could *physically* call a particular round ball of shiny metal 'steel', but that would not *legally* mean the item is called 'steel, according to Clause D.1' as the conformance constraints are not met. Subjective interpretations of words or clauses are dealt with in [AB2002a, Abr2002].

[5] 'Invitation to treat' is a construct of English law intended to capture a non-binding suggestion by a party. Acceptance of a legal offer creates an obligation, whereas acceptance of an invitation to treat has no such result [TB99]. Similarly, accepting a volunteer does not lead to the creation of an obligation under English law, since volunteering implies there is no expectation of payment and, except for the special case of deeds, English law requires the existence of consideration (exchange) for the formation of a valid contract [TB99].

[6] Austin [Aus76] and Searle [Sea69] use the term 'illocutionary force' rather than 'attitude'.

Power is encoded in a variety of ways. Nouns, verbs, adjectives and adverbs often have clause-specific meanings, and imply the existence of criteria for items or occurrences in that class. Also, the English conditionals 'if', 'if … then …', and 'when' may indicate that certain occurrences only come about upon the existence of particular conditions, which are described in terms of other occurrences. Similarly, the English prepositions 'to', 'in order to', 'by', and 'through' may indicate that some set of occurrences brings about some other occurrence.	**Guideline 15**

Examples:

By Guideline 15, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Power of Type
The steel is returned if/when it is shipped to the registered address of Steelmans		An act of (successful) *shipping* brings about an occurrence of *returning*
The steel is returned by/through shipping it to the registered address of Steelmans		"
To / In order to return the steel, ship it to the registered address of Steelmans		"

A conventional occurrence of 'returning' is recognized when occurrences fitting a description ('shipping to the registered address of Steelmans') happen. This is one example of a power (here, of Steelmans to return steel) encoded in a contract.

The English conditionals 'provided', 'providing', 'on condition that', 'as long as', 'so long as', 'unless', and 'in order to', may indicate minimal necessary conditions that must exist in order for a state of affairs to come about. That is, an *immunity* [AB2002a, Abr2002], or alternatively a *voidance* [AB2002b, AEB2002c], is encoded: unless those conditions exist, the state of affairs is not taken to obtain.	**Guideline 16**

Examples:

By Guideline 16, (Provenance)

(Evidence)		(Consequence)
Appearance of Word	… indicates …	Immunity of Type / Voidance
The steel may be returned provided / as long as / on condition that it is undamaged.		if the steel is damaged, then (according to this clause), `count (returnings of this steel) = 0`, or, alternatively, all obligations resulting from this return are *voided*.
The steel may be returned unless it is damaged.		"
In order to return the steel, it must be undamaged.		"

Caution should, as always, be used in applying these guidelines. For instance, 'in order to' may indicate intention, expectation, or recommendation (occurrences of *intending*, *expecting*, or *recommending*), rather than existence of legal power or a guarantee. For example, in …

- The steel may be trucked by CargoCarriers in order to return it.

… the specification certainly seems to indicate a permission (based on the appearance of 'may'). However, it is not clear whether it is meant to confer a *power* to return steel via that means, or merely gives a *recommendation* as to how steel could be returned. Less forcefully even, it might simply give an *expectation* as to how the goal might be achieved. That is, it might just say that occurrences of trucking steel back with CargoCarriers are intended or expected to result in occurrences of returning the steel. Disambiguation of these various senses is important in contracts, as a party may wish to recommend a course of action to another party, but may not wish to confer a legal power to the other party. Specifying explicitly that this is a recommendation or expectation, rather than a conferral of power or guarantee, may reduce misunderstanding and avoid contractual disputes.

5 Conclusion and Future Work

This paper has introduced and documented a spectrum of techniques for exposing occurrences in an English language specification. The intention was to provide insights that may be employed by an analyst when attempting to uncover logical structures in natural language specifications. Though we have attempted to be thorough, the rules we have set out are by no means exhaustive, and we have also not considered counterexamples in any depth. The intention was simply to provide helpful insights that may be employed by an analyst when uncovering logical structures in natural language specifications. We have shown how to move from varying expressions of semantics in English to canonical forms – for example, various ways of expressing negation in English, such as 'no' and 'un–', can be rendered as the canonical form '*counting*, with `count=0`'; and diverse prescriptive expressions like 'must', 'have to', and '–able' might be rendered as implying occurrences of *being-obliged*.

We qualified our proposals with the reminder that, with the current state-of-the-art in natural language understanding technology, the process cannot hope to be mechanistically systematic. Each of the indicative keywords and fragments we have mentioned point to the existence of instances of certain types of occurrences. We seek here only to capture a subset of meanings that may be useful in enlightening the legal and semantic structure of an English language specification of contracts, policies, or regulations. Our guidelines are useful in that they make explicit some helpful rules that analysts can employ when developing structured representations of natural language specifications. Storage, consistency checking, and

execution of such structured specifications have been demonstrated elsewhere [AB2001c, AB2002a, Abr2002, AEB2002a, AEB2002b, AEB2002c]. The goal in this paper has been to move some way towards codifying the hitherto unsystematic transition from analysis to implementation.

Implementation of our prototype software in a distributed setting, by providing facilities for distributing and collating provisions and occurrences remains to be addressed; currently we store this information in a centralized database. We also plan to enhance usability by defining high-level business contract definition templates at user-interface level, and providing completeness checks.

References

[AB2001b] Abrahams AS and Bacon JM. Occurrence-centric Policy Specification for E-commerce Applications. *Workshop on Formal Modelling for Electronic Commerce (FMEC 2001).* Oslo, Norway. June 2001.

[AB2001c] Abrahams AS and Bacon JM. Representing and Enforcing Electronic Commerce Contracts over a Wide Range of Platforms Using Occurrence Stores. *4th CaberNet Plenary Workshop*. Pisa, Italy. October 2001.

[AB2001d] Abrahams AS and Bacon JM. Representing and Enforcing E-Commerce Contracts Using Occurrences. *Proceedings of the 4th International Conference on Electronic Commerce Research (ICECR4).* Dallas, TX. November 2001.

[AB2002a] Abrahams AS and Bacon JM. A Software Implementation of Kimbrough's Disquotation Theory for Representing and Enforcing Electronic Commerce Contracts. *Group Decision and Negotiations Journal.* 11(6). Special Issue on Formal Modelling in Electronic Commerce. INFORMS. pp. 1-38. November 2002.

[AB2002b] Abrahams AS and Bacon JM. The Life and Times of Identified, Situated, and Conflicting Norms. *Sixth International Workshop on Deontic Logic in Computer Science (DEON'02).* Imperial College, London, UK. May 2002.

[Abr2002] Abrahams AS. *Developing and Executing Electronic Commerce Applications with Occurrences.* PhD Thesis. University of Cambridge Computer Laboratory. Cambridge, UK. September 2002.

[AEB2002a] Abrahams AS, Eyers DM, and Bacon JM. A Coverage Determination Mechanism for Checking Business Contracts against Organizational Policies. *3rd VLDB Workshop on Technologies for E-Services (TES'02).* Hong Kong, China. August 2002. Lecture Notes in Computer Science 2444. Springer-Verlag. pp. 97-106. 2002.

[AEB2002b] Abrahams AS, Eyers DM, and Bacon JM. Mechanical Consistency Analysis for Business Contracts and Policies. *Proceedings of the 5th International Conference on Electronic Commerce Research (ICECR5).* Montreal, Canada. October 2002.

[AEB2002c] Abrahams AS, Eyers DM, and Bacon JM. An asynchronous rule-based approach for business process automation using obligations. *Proceedings of the 3rd ACM SIGPLAN Workshop on Rule-Based Programming (RULE'02).* Pittsburgh, PA. October 2002.

[AK2002] Abrahams AS and Kimbrough SO. Treating Disjunctive Obligation and Conjunctive Action in Event Semantics with Disquotation. *Wharton Business School Working Paper Series*. University of Pennsylvania. 2002.

[Aus76] Austin JL. *How to do things with words.* Oxford University Press. 1976.

[BLM2001] Bacon J, Lloyd M, and Moody K. Translating role-based access control policy within context. *Proceedings of Policies for Distributed Systems and Networks: International Workshop (Policy 2001).* Bristol, UK. Lecture Notes in Computer Science 1995. Springer-Verlag. January 2001. pp. 107–119.

[Dav80] Davidson D. *Essays on Actions and Events.* Clarendon Press, Oxford. 1980.

[HvdVH97] Hoppenbrouwers J, van der Vos B, and Hoppenbrouwers S. NL Structures and Conceptual Modelling: Grammalizing for KISS. *Data and Knowledge Engineering.* 23(1). Elsevier Science Publishers. Amsterdam, The Netherlands. pp. 79-92. June 1997.

[JS96] Jones AJI and Sergot M. A formal characterisation of institutionalised power. *Journal of the Interest Group in Pure and Applied Logic.* 4(3). pp. 427-443. 1996.

[KM97] Kimbrough SO and Moore SA. On Automated Message Processing in Electronic Commerce and Work Support Systems: Speech Act Theory and Expressive Felicity. *ACM Transactions on Information Systems.* 15(4). ACM Press. pp. 321-367. October 1997.

[Llo2000] Lloyd MS. *Conversion of Access Control Policy to Formal Logic.* MPhil Thesis. University of Cambridge Computer Laboratory. 2000.

[MOR2001] Michael JB, Ong VL, and Rowe NC. Natural Language Processing Support for Developing Policy-Governed Software Systems. *39th International Conference on Object-Oriented Languages and Systems (TOOLS USA 2001).* Santa Barbara, California. July 2001.

[OM96] Osborne M and MacNish C. Processing natural language software requirements specifications. *International Conference on Requirements Engineering.* Colorado Springs. 1996

[Par90] Parsons T. *Events in the Semantics of English: A Study in Subatomic Semantics.* MIT Press. Cambridge, MA. 1990.

[Sea69] Searle JR. *Speech acts: An essay in the philosophy of language.* Cambridge University Press. 1969.

[Sta2002] Staskiewicz D. *Logikbasierte Überwachung der Vertragserfüllungsphase im elektronischen Handel.* Diplomarbeit Informatik am Lehrstuhl IV der RWTH Aachen. Draft. Aachen, Germany. December 2002.